Udacity Artificial Intelligence Nanodegree

Build a Forward-Planning Agent project

# Data analysis

## Problem 1

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Search Algorithm | Heuristic | Domain Actions | Expansions | Goal Tests | New Nodes | Plan Length | Time elapsed (seconds) |
| Breadth First Search |  | 20 | 43 | 56 | 178 | 6 | 0.02634 |
| Depth First Search |  | 20 | 21 | 22 | 84 | 20 | 0.01680 |
| Uniform Cost Search |  | 20 | 60 | 62 | 240 | 6 | 0.06567 |
| Greedy Best First Search | Unmet Goals | 20 | 7 | 9 | 29 | 6 | 0.01040 |
| Level Sum | 20 | 6 | 8 | 28 | 6 | 0.47710 |
| Max Level | 20 | 6 | 8 | 24 | 6 | 0.36520 |
| Set Level | 20 | 6 | 8 | 28 | 6 | 1.05531 |
| A\* Search | Unmet Goals | 20 | 50 | 52 | 206 | 6 | 0.05627 |
| Level Sum | 20 | 28 | 30 | 122 | 6 | 0.70481 |
| Max Level | 20 | 43 | 45 | 180 | 6 | 0.78900 |
| Set Level | 20 | 33 | 35 | 138 | 6 | 1.70491 |

## Problem 2

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Search Algorithm | Heuristic | Domain Actions | Expansions | Goal Tests | New Nodes | Plan Length | Time elapsed (seconds) |
| Breadth First Search |  | 72 | 3343 | 4609 | 30503 | 9 | 0.64033 |
| Depth First Search |  | 72 | 624 | 625 | 5602 | 619 | 1.09864 |
| Uniform Cost Search |  | 72 | 5154 | 5156 | 46618 | 9 | 1.16339 |
| Greedy Best First Search | Unmet Goals | 72 | 17 | 19 | 170 | 9 | 0.06380 |
| Level Sum | 72 | 9 | 11 | 86 | 9 | 1.26094 |
| Max Level | 72 | 27 | 29 | 249 | 9 | 1.65193 |
| Set Level | 72 | 9 | 11 | 84 | 9 | 3.96944 |
| A\* Search | Unmet Goals | 72 | 2467 | 2469 | 22522 | 9 | 1.35928 |
| Level Sum | 72 | 357 | 359 | 3426 | 9 | 12.34326 |
| Max Level | 72 | 2887 | 2889 | 26594 | 9 | 69.45395 |
| Set Level | 72 | 1037 | 1039 | 9605 | 9 | 186.80754 |

## Problem 3

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Search Algorithm | Heuristic | Domain Actions | Expansions | Goal Tests | New Nodes | Plan Length | Time elapsed (seconds) |
| Breadth First Search |  | 88 | 14663 | 18098 | 129625 | 12 | 0.88557 |
| Depth First Search |  | 88 | 408 | 409 | 3364 | 392 | 0.27942 |
| Uniform Cost Search |  | 88 | 18510 | 18512 | 161936 | 12 | 1.37449 |
| Greedy Best First Search | Unmet Goals | 88 | 25 | 27 | 230 | 15 | 0.04219 |
| Level Sum | 88 | 14 | 16 | 126 | 14 | 1.32965 |
| Max Level | 88 | 21 | 23 | 195 | 13 | 1.20817 |
| Set Level | 88 | 35 | 37 | 345 | 17 | 7.75036 |
| A\* Search | Unmet Goals | 88 | 7308 | 7390 | 65711 | 12 | 1.11634 |
| Level Sum | 88 | 369 | 371 | 3403 | 12 | 12.83952 |
| Max Level | 88 | 9580 | 9582 | 86312 | 12 | 224.31976 |
| Set Level | 88 | 3423 | 3425 | 31596 | 12 | 536.24249 |

## Problem 4

|  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- |
| Search Algorithm | Heuristic | Domain Actions | Expansions | Goal Tests | New Nodes | Plan Length | Time elapsed (seconds) |
| Breadth First Search |  | 104 | 99736 | 114953 | 944130 | 14 | 4.77090 |
| Depth First Search |  | 104 | 25174 | 25175 | 228849 | 24132 | 832.20874 |
| Uniform Cost Search |  | 104 | 113339 | 113341 | 1066413 | 14 | 7.71296 |
| Greedy Best First Search | Unmet Goals | 104 | 29 | 31 | 280 | 18 | 0.04816 |
| Level Sum | 104 | 17 | 19 | 165 | 17 | 1.54115 |
| Max Level | 104 | 56 | 58 | 580 | 17 | 2.45123 |
| Set Level | 104 | 107 | 109 | 1164 | 23 | 29.44411 |
| A\* Search | Unmet Goals | 104 | 34330 | 34332 | 328509 | 14 | 4.46616 |
| Level Sum | 104 | 1208 | 1210 | 12210 | 15 | 64.66808 |
| Max Level | 104 | 62077 | 62079 | 599376 | 14 | 2132.38761 |
| Set Level | 104 | 22606 | 22608 | 224229 | 14 | 5914.70926 |

# Conclusions

* **Which algorithm or algorithms would be most appropriate for planning in a very restricted domain (i.e., one that has only a few actions) and needs to operate in real time?**

Given a very restricted domain, Greedy Best First Search with Unmet Goals heuristic and Breadth First Search seem to do better than the others for problems 1 and 2.

* **Which algorithm or algorithms would be most appropriate for planning in very large domains (e.g., planning delivery routes for all UPS drivers in the U.S. on a given day)**

Breadth First Search and A\* Search with Unmet Goals heuristic seem to be the most appropriate. After them, Uniform Cost Search seems to do a better job than the other strategies.

* **Which algorithm or algorithms would be most appropriate for planning problems where it is important to find only optimal plans?**

Breadth First Search, Uniform Cost Search and A\* Search with Unmet Goals heuristic find optimal plans for all the observed problems.